

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-221399

(43)Date of publication of application : 30.08.1996

(51)Int.CI. G06F 17/21
G06F 17/27
G06F 17/24

(21)Application number : 07-029096

(71)Applicant : HITACHI LTD

(22)Date of filing : 17.02.1995

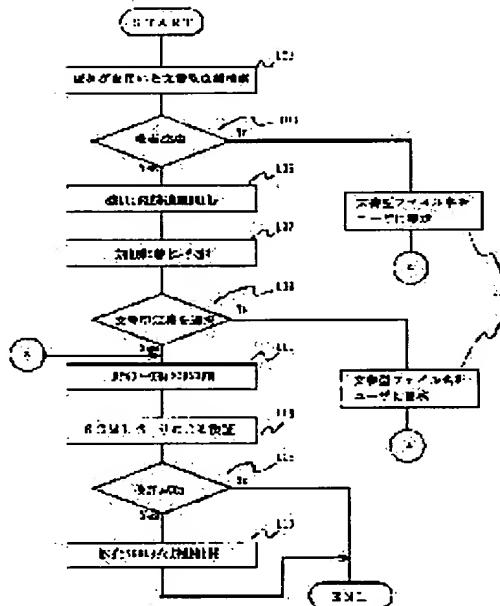
(72)Inventor : SATO YOSHIFUMI
HINO MASATOSHI

(54) METHOD AND DEVICE FOR STRUCTURED DOCUMENT PREPARATION

(57)Abstract:

PURPOSE: To check the matching degree of a source text and respective document logical structure definitions and to easily prepare a document instance corresponding to a structured document form by making user definition tag information and stipulated tag information inside the respective document logical structure definitions correspond to each other beforehand.

CONSTITUTION: By referring to the correspondence relation of the user definition tag information in the source text and the document logical structure definitions of a change object, the matching degree of the source text and the respective document logical structure definitions is checked 101. In the order of the matching degree of the user definition tag information and the document logical structure definitions, the retrieved document logical structure definitions are presented to the user 107. By letting a user select the document logical structure definition, the document logical structure definition to be a target is decided from the presented plural document logical structure definitions 109. Matched with the definition selected by the user, the user definition tag information in the source text is converted into the stipulated tag information 111. By verifying 113 the converted text by a parser for performing syntax analysis, whether or not the document instance is in alignment with the document logical structure definition is judged.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平8-221399

(43)公開日 平成8年(1996)8月30日

(51) Int.Cl. ⁶ G 0 6 F 17/21 17/27 17/24	識別記号 9288-5L 9288-5L 9288-5L	序内整理番号 F I G 0 6 F 15/20	技術表示箇所 5 8 6 A 5 5 0 E 5 5 8
--	---------------------------------------	--------------------------------	---------------------------------------

審査請求 未請求 請求項の数6 OL (全16頁)

(21)出願番号 特願平7-29096

(22)出願日 平成7年(1995)2月17日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 里 佳史

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

(72)発明者 鎌野 匡利

神奈川県川崎市麻生区王禅寺1099番地 株式会社日立製作所システム開発研究所内

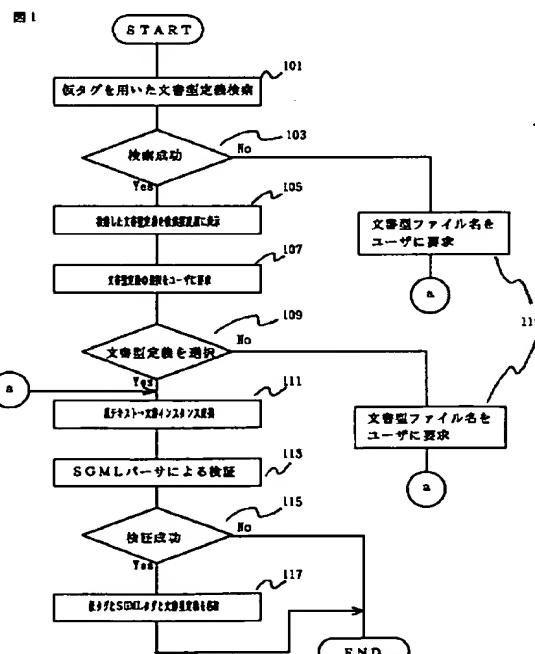
(74)代理人 弁理士 小川 勝男

(54)【発明の名称】 構造化文書作成方法および装置

(57)【要約】 (修正有)

【構成】 予め用意したユーザが定義した文書論理構造を表す情報（仮タグ）と複数の文書論理構造定義（文書型定義）との対応関係を参照して、ユーザが仮タグを用いて作成した原テキストと各文書型定義との一致度（検索頻度）を求め101、その一致度順に文書型定義を表示し、ユーザに対して文書型定義の選択要求を行う。更に、ユーザの選択した文書型定義109に合わせて、原テキスト中の仮タグを、文書型定義に基づく文書論理構造を表す情報（SGMLタグ情報）に変換し、変換結果を含むテキストを、文書型定義に基づいて構文解析を行なうパーサ（SGMLパーサ）によって解析し、変換結果の整合性の検証を行なう113。

【効果】 複数の文書型定義の中から、ユーザが仮タグを用いて作成した原文書に対応する文書型定義の選択及び文書型定義に応じた文書インスタンスへの変換を容易に行なうことができる。



【特許請求の範囲】

【請求項 1】文書から所定の文書論理構造定義にしたがった構造化文書を作成する構造化文書作成方法において、
予め文書の論理構造を示す所定の論理構造定義を複数記憶しておき、
上記文書についてユーザが定義した論理構造を示す情報と、記憶された所定の文書論理構造定義を比較して、一致度を求め、
比較された文書論理構造を一致度順に表示し、
表示された文書論理構造定義を上記ユーザが選択し、
上記ユーザの選択に基づいて上記ユーザが定義した文書の論理構造を示す情報をユーザの選択した文書論理構造定義で用いられる文書の論理構造を示す情報に変換することを特徴とする構造化文書作成方法。

【請求項 2】請求項 1 に記載の構造化文書作成方法において、
上記ユーザが定義した文書論理構造を示す情報を用いて行なう複数の文書論理構造定義との対応付けを、少なくともユーザが定義した文書論理構造を示す情報と、文書論理構造定義の名称又は文書論理構造定義を一意に示す情報と、各文書論理構造定義においてテキスト中で用いる文書論理構造を示す情報の三項関係を持つ関係テーブルを用いて行なうことを特徴とする構造化文書作成方法。

【請求項 3】請求項 2 に記載の構造化文書作成方法において、
上記ユーザが定義した文書論理構造を示す情報を登録された文書論理構造定義で用いられる文書論理構造を示す情報を変換した後、変換された文書論理構造定義を含む文書が文書論理構造定義に適合するかどうかを検証することを特徴とする構造化文書作成方法。

【請求項 4】請求項 2 に記載の構造化文書作成方法において、
上記関係テーブルに登録内容の過不足を確認し、
確認の結果不足が存在することが検出された場合に、ユーザが定義した文書論理構造を示す情報、文書論理構造定義の名称又は文書論理構造定義を一意に示す情報、文書論理構造定義で用いられる文書論理構造を示す情報、の三項目を新たに対応付け、上記三項関係を持つ関係テーブルに登録することを特徴とする構造化文書作成方法。

【請求項 5】請求項 1 に記載の構造化文書作成方法において、
上記文書中のユーザが定義した文書論理構造を示す情報、及び、文書論理構造定義で用いられる文書論理構造を示す情報の一方、又は、両方を用いて文書論理構造定義との対応付けを行なうことを特徴とする構造化文書作成方法。

【請求項 6】文書から所定の文書論理構造定義にしたが

った構造化文書を作成する構造化文書作成装置において、
予め文書の論理構造を示す所定の論理構造定義を複数記憶する手段と、
上記文書についてユーザが定義した論理構造を示す情報と、記憶された所定の文書論理構造定義を比較し、一致度を求める手段と、
比較された文書論理構造を一致度順に表示する手段と、表示された文書論理構造定義を上記ユーザが選択する手段と、
上記ユーザの選択に基づいて上記ユーザが定義した文書の論理構造を示す情報をユーザの選択した文書論理構造定義で用いられる文書の論理構造を示す情報に変換する手段を有することを特徴とする構造化文書作成装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、ユーザが作成した文書から所定の文書論理構造定義に従った構造化文書を作成する構造化文書作成方法および装置に関するものである。

【0002】

【従来の技術】構造化文書の形式の一つに、論理構造を明示的に表す情報をテキスト中に埋め込むものがある。一般にユーザが作成した構造化文書（以下、文書インスタンスと呼ぶ）は、文書の論理構造を規定する文書論理構造定義を記述したファイルを指定する部分と、文書の内容を表す内容テキスト部からなることが多い。文書論理構造定義には、その文書の論理構造と、その構成要素を表わすマーク（以下、タグと呼ぶ）が定義されている。また、内容テキスト部には、文書論理構造定義内で定義されたタグを、そのタグに対応する論理構造の内容となる文字列が一意に定まるように挿入し、文書の論理構造を明示的に表現する。

【0003】文書インスタンスは、一般にテキストエディタや、構造化文書専用のエディタである構造化文書エディタを用いて作成する。また、文書インスタンス中のタグが正しく付けられているか否かは、指定された文書論理構造定義に従って文書インスタンスの構文解析を行うバーサによって検証される。

【0004】他に文書インスタンスを生成する手段としては、ワードプロセッサ等で作成した、タグのない文書（以下、ブレーンテキストと呼ぶ）の文書論理構造を解析して、構造化文書の文書インスタンスを作成するという手段も考えられる。この手段に必要なブレーンテキストの論理構造を解析する技術として特開昭62-249270号公報に記載された「文書処理装置」がある。この装置では、見出し語辞書、見出し規則辞書、文章構造規則辞書等を参照して文書構造を判定し、この文書構造に従って前記文書データの、例えば、章、節、箇条書、段落などの階層的な論理構造を求めるものである。

【0005】

【発明が解決しようとする課題】近年、組織内に氾濫する膨大な量の文書を電子的かつ統一的に扱うため、組織内の文書を全て構造化文書形式で蓄積し、共有／活用する文書情報処理システムの構築が活発化している。また、米国防総省のCALS(Continuous Acquisition and Lifecycle Support)において、国際標準構造化文書形式SGMLをベースとして規定された形式の文書のみ受け付けるという例に見られるように、文書情報処理システムへのデータ入力に要する労力を削減するため、受け付ける文書を特定の構造化文書形式に限定する（あるいはその形式に従った文書を優遇する）という動きもある。

【0006】しかし、対象とする構造化文書形式を扱い慣れない者にとって、最初から正確な構造化文書を作成することは困難である。これは、以下に示すような理由による。

【0007】まず、テキストエディタを用いて文書インスタンスを作成する場合、「対象とする文書に対してどのような文書論理構造定義が用意されており、どのようにタグを付けるべきか」というような、文書インスタンスの記述方法に関する正確な知識をユーザが持つ必要がある。

【0008】また、構造化文書エディタを用いて文書インスタンスを作成する場合には、扱い慣れないタグを付加する労力を軽減することが出来るが、ユーザが使用する計算機環境に構造化文書エディタが存在する必要がある。

【0009】さらに、先に挙げた特開昭62-249270号公報における、ブレーンテキストから論理構造を抽出する方法については、予め与えられた見出し語や文書構造に従って文書論理構造を求めるため、与えられていない見出し語や文書構造については対応できない。

【0010】以上の理由により、従来の技術では、対象とする構造化文書形式を扱い慣れない者にとって、その形式に沿った正確な文書インスタンスを作成することは困難である。

【0011】

【課題を解決するための手段】本発明は、ユーザが独自に定義した文書の論理構造を示す情報（以下、ユーザ定義タグ情報と呼ぶ）と、対象とする構造化文書形式において予め用意されている複数の文書論理構造定義とを対応付けてその一致度を求める手段と、ユーザがユーザ定義タグ情報を用いて作成した原テキストについて文書論理構造定義との一致度を求め、一致度順に文書論理構造定義を表示する手段と、表示された文書論理構造定義をユーザが選択する手段と、ユーザの選択した文書論理構造定義に合わせて原テキスト中のユーザ定義タグ情報を文書論理構造定義で用いられている文書の論理構造を示す情報（以下、規定タグ情報と呼ぶ）に変換する手段とを具備して構成するものである。

【0012】ユーザ定義タグ情報と文書論理構造定義との対応付けは、例えばユーザ定義タグ情報と、文書論理構造定義名称又は文書論理構造定義を一意に示す情報と、規定タグ情報との三項関係を持つ関係テーブルを用いて行う。

【0013】また、上記関係テーブルの登録内容の不足が検出される場合に、未登録の三項関係を上記関係テーブルに登録する手段を設ける。

10 【0014】さらに、変換結果を含むテキストが文書論理構造定義に適合するか否かを検証する手段（バーサ）を設ける。

【0015】なお、ユーザの作成する原テキストは、ユーザ定義タグ情報及び規定タグ情報の一方、又は両方を用いて記述して良いとする。

【0016】本発明では、文書の目的にあった文書論理構造定義を検索する手段を提供することで、予め決められた論理構造にユーザの作成する原テキストを合わせるのではなく、原テキストに合った文書論理構造定義を選択し、その文書論理構造定義に沿った構造化文書を生成することができる。

【0017】

【作用】原テキスト中のユーザ定義タグ情報と、変換対象の文書論理構造定義との対応関係を参照することにより、原テキストと各文書論理構造定義との一致度を調べる。

【0018】ユーザ定義タグ情報と文書論理構造定義との一致度順に、検索された文書論理構造定義をユーザに対して提示する。

30 【0019】ユーザが文書論理構造定義の選択を行うことで、提示された複数の文書論理構造定義の中から目的とする文書論理構造定義が決定する。

【0020】ユーザの選択した文書論理構造定義に合わせて原テキスト中のユーザ定義タグ情報を規定タグ情報に変換する。その過程でタグ情報間の対応関係の不足が検出された場合には、新たな対応関係の関係テーブル登録をユーザに対して要求する。

40 【0021】変換したテキストを、文書論理構造定義に従って構文解析を行うバーサによって検証することで、文書インスタンスが文書論理構造定義沿っているか否かが判定される。これによりユーザ定義タグ情報から規定タグ情報への変換が正しく行われたか否かの判定が行われる。

【0022】以上述べたように、ユーザの馴染みやすいユーザ定義タグ情報と各文書論理構造定義内の規定タグ情報を予め対応付けておくことにより、ユーザはユーザ定義タグ情報のみを用いて原テキストを作成すればよく、文書論理構造定義名称や、その定義に従った記述方法を正確に把握しておく必要はない。そのため、対象とする構造化文書形式を扱い慣れない者にとっても、容易にその文書形式に従った文書インスタンスを作成するこ

50

5
とが可能になる。

【0023】

【実施例】本発明の一実施例を図を用いて説明する。本実施例においては、構造化文書形式としてSGML形式を採用し、文書論理構造定義に相当する概念はSGMLの「文書型定義」であるとする。ここでは、本発明を用いた構造化文書作成プログラム301について説明する。

【0024】図1は、構造化文書作成プログラム301の処理フローである。この処理フローを説明する前に、ハードウェア構成とシステム構成を図2と図3を用いて説明する。

【0025】図2はハードウェア構成を示す図である。ハードウェアは主記憶装置201、CPU203、二次記憶装置205、キーボード207、マウス209、CRT211を持つ計算機である。

【0026】図3はシステム構成を示す図である。

【0027】構造化文書作成プログラム301は原テキストファイル307から原テキストを読み込み、関係テーブルファイル309にある関係テーブル600を用いて原テキストを文書インスタンスに変換して、結果ファイル311に出力する。

【0028】SGMLバーサ305は文書型定義ファイル313中の文書型定義を用いて結果ファイル311中の文書インスタンスの検証を行う。

【0029】連絡領域303は構造化文書作成プログラム301とSGMLバーサ305とのデータのやりとりを行うための領域である。

【0030】原テキストファイル307の内容を図4に、結果ファイル311の内容である文書インスタンスを図5に、関係テーブルファイル309の内容を図6にそれぞれ示す。

【0031】図4は原テキストの内容を示す図である。

【0032】原テキストはユーザが定義したタグ（以下、仮タグと呼ぶ）を付けられた文書である。

【0033】仮タグには開始仮タグ（401、403、405、407、409）と終了仮タグ（402、404、406、408、410）がある。開始仮タグは記号“<<”と記号“>>”で囲まれており、記号で挟まれた文字列が開始仮タグ名称である。終了仮タグは記号“<</”と記号“>>”で囲まれており、記号で挟まれた文字列が終了仮タグ名称である。開始仮タグと終了仮タグで囲まれたテキストが仮タグで示される内容である。

【0034】図5は文書インスタンスを示す図である。

【0035】文書インスタンスは文書型定義情報とタグ付けされた文書からなる。

【0036】文書型定義情報は文書インスタンスの型を定義する情報であり、図5では文書型宣言文501で外部ファイルに書かれた型を定義している。

6

【0037】文書インスタンスはSGMLタグを付けられた文書である。SGMLタグには開始タグ（502、504、506、508、510）と終了タグ（503、505、507、509、511）がある。開始タグは記号“<”と記号“>”で囲まれており、記号で挟まれた文字列が開始タグ名称である。終了タグは記号“</”と記号“>”で囲まれており、記号で挟まれた文字列が終了タグ名称である。開始タグと終了タグで囲まれたテキストがタグで示された内容である。

10 【0038】図6は関係テーブル600である。関係テーブル600は仮タグ欄601、SGMLタグ欄602、文書型定義ファイル名称欄603からなる。関係テーブル600の各行は仮タグとSGMLタグと文書型定義の三項関係を表す。

【0039】仮タグ欄601は原テキスト中に記述された仮タグを格納する。SGMLタグ欄602は文書インスタンス中のSGMLタグを格納する。文書型定義ファイル名称欄603は文書型定義ファイル名称を格納する。

20 【0040】次に図1の処理について説明する。

【0041】処理101は仮タグを用いた文書型定義検索処理である。処理101は原テキスト中の仮タグから文書型定義ファイル名を検索する処理である。処理101の詳細を図8を用いて説明する。また、処理101の別の処理方法を図9を用いて説明する。

【0042】図8は処理101の詳細処理フローである。

30 【0043】処理801は原テキストから仮タグを1件抜き出す処理である。原テキスト中の仮タグは記号“<<”と記号“>>”で囲まれているので、記号“<<”と記号“>>”の記号に注目して仮タグを抜き出す。

【0044】処理803は原テキスト中から全てのタグを抜き出したか否かを判定する処理である。全ての仮タグを抜き出した場合は処理811に移り、原テキスト中にまだ抜き出していない仮タグがある場合には処理805に移る。

【0045】処理805は処理801で抜き出した仮タグを用いて関係テーブル600の仮タグ欄601を検索し、該当する関係を取り出す処理である。

40 【0046】処理807は処理805で仮タグの検索が成功したか否かの判定を行う処理である。検索が成功した場合は処理809に移り、検索が失敗した場合は処理801に移る。

【0047】処理809は処理805の検索でヒットした関係の中の文書型定義ファイル名称の頻度を記憶する処理である。この検索時の頻度情報（以下、検索頻度と呼ぶ）が、原テキストと文書型定義との「一致度」に相当する概念である。

50 【0048】処理811は処理809で文書型定義ファイル名を記憶したか否かを判定する処理である。1件で

も記憶していれば処理101は成功（処理813）とし、記憶してなければ処理101は失敗（処理815）とする。

【0049】図9は処理101の別法の詳細処理フローである。

【0050】図9の処理方法は原テキスト中に仮タグだけでなくSGMLタグが混在している場合に、SGMLタグをも文書型定義検索に利用するための方法である。

【0051】処理901は原テキストから仮タグまたはSGMLタグを1件抜き出す処理である。原テキスト中の仮タグは記号”<<”と記号”>>”で囲まれているので、記号”<<”と記号”>>”の記号に注目して仮タグを抜き出す。また原テキスト中のSGMLタグは記号”<”と記号”>”で囲まれているので記号”<”と記号”>”の記号に注目して抜き出す。

【0052】処理903は原テキスト中から全てのタグを抜き出したか否かを判定する処理である。全ての仮タグまたはSGMLタグを抜き出したときは処理915に移り、原テキスト中にまだ仮タグまたはSGMLタグがあるときには処理905に移る。

【0053】処理905は処理901で抜き出したタグの種類を判定する処理である。抜き出したタグが仮タグの場合は処理907に移り、抜き出したタグがSGMLタグの場合は処理909に移る。

【0054】処理907は処理901で抜き出した仮タグを用いて関係テーブル600の仮タグ欄601を検索し、関係を取り出す処理である。

【0055】処理909は処理901で抜き出したSGMLタグを用いて関係テーブル600のSGMLタグ欄602を検索し、関係を取り出す処理である。

【0056】処理911は処理907で仮タグの検索が、また処理909でSGMLタグの検索が成功したか否かの判定を行う処理である。検索が成功したときは処理913に移り、検索が失敗したときは処理901に移る。

【0057】処理913は処理907または処理909で検索した関係中の文書型定義ファイル名称の頻度を記憶する処理である。

【0058】処理915は処理913で文書型定義ファイル名を記憶したか否かを判定する処理である。1件でも記憶していれば処理101は成功（処理917）とし、記憶してなければ処理101は失敗（処理919）とする。

【0059】処理103は処理101が成功したか否かを判定する処理である。処理101が成功ならば処理105に移り、失敗ならば処理119に移る。

【0060】処理105は処理101で検索した文書型定義を検索頻度順に表示する処理である。処理105の詳細を図10を用いて説明する。

【0061】図10は検索した文書型定義を検索頻度順

に表示する処理の詳細フローである。

【0062】処理1001は処理101で検索し記憶した文書型定義ファイル名称を検索頻度の降順にソートする処理である。降順にソートすることで文書型定義ファイル名称を検索頻度の高い順に順序付ける。

【0063】処理1003は処理1001でソートした文書型定義ファイル名称の中から検索頻度の高い文書型定義ファイル名称から順に1件取り出す処理である。

【0064】処理1005は処理1003で取り出した文書型定義ファイル名称とそのファイルの内容を表示する処理である。

【0065】処理1007は次に頻度の高い文書型定義ファイル名称が存在するか否かを判定する処理である。存在すれば処理1001に移り、存在しなければ処理105を終了する。

【0066】処理105の表示内容を図11に示す。

【0067】図11は文書型定義のファイル名と内容の表示を示す図である。

【0068】枠1100は文書型定義のファイル名と内容を表示する表示枠である。枠1100には処理101で検索した全ての文書型定義について表示する。

【0069】枠1101には処理101で検索した全ての文書型定義ファイル名を表示する。

【0070】枠1102には処理101で検索した全ての文書型定義ファイルの内容を表示する。

【0071】枠1103と枠1105はそれぞれ1件の文書型定義ファイルについてその名称と内容を表示する表示枠である。

【0072】枠1104と枠1106はそれぞれ1件の文書型定義ファイル名称を表示する表示枠である。枠1107（枠1109）は枠1104（枠1106）に示される文書型定義ファイルの内容を表示する表示枠である。

【0073】マーク1110、マーク1111は表示枠1100をスクロールするボタンである。

【0074】マーク1112、マーク1113（マーク1114、マーク1115）は表示枠1107（表示枠1109）をスクロールするボタンである。

【0075】処理107は文書型定義の選択をユーザに要求する処理である。文書型定義の選択の手段を図12を用いて、また処理の詳細を図13を用いて説明する。

【0076】図12は文書型定義の選択画面である。文書の選択は図11で示した枠1100を用いて行う。

【0077】図12において、マーク1200はマウスカーソルであり、マウス209を用いて動かされる。枠1201は空の表示枠である。

【0078】図13は処理107の詳細フローである。

【0079】処理1301はユーザの入力待ちを行う処理である。ユーザ入力待ちはマウス209のボタンをユーザが押すまで続く。

【0080】処理1303は処理1301でマウス209のボタンが押された位置を判定する処理である。空の表示枠が選択された場合は文書型定義の選択は行われなかつこととする(処理1304)。文書型定義ファイル名称が表示されている枠を選択した場合は選択した枠の文書型定義の選択は行われたこととする(処理1305)。スクロールボタン(1110、1111、1112、1113、1114、1115)を選択した場合は該当する枠のスクロール処理(処理1307)を行う。

【0081】処理109は処理107で文書型定義を選択したか否かを判定する処理である。

【0082】処理107で文書型定義を選択しなかつたときは処理119に移り、文書型定義を選択したときは処理111に移る。

【0083】処理111は原テキストを文書インスタンスに変換する処理である。処理111の詳細処理を図14を用いて説明する。

【0084】図14は処理111の詳細フローである。

【0085】処理1401は処理107または処理119で決まった文書型定義ファイルに含まれる論理構造の中から、目的とする構造の名称をユーザに要求する処理である。ユーザは処理107または処理119で選択した文書型定義ファイル中の目的とする論理構造の名称を入力する。

【0086】処理1403は処理107または処理119で決まった文書型定義ファイル名称と処理1401で入力した論理構造名称から文書型宣言文501を生成する。処理1405は処理1405で生成された文書型宣言文501を結果ファイル311に出力する。

【0087】処理1407は、関係テーブル600に記述された三項関係に従って、原テキスト中の仮タグを処理107または処理119で決まった文書型定義で定義されたSGMLタグに変換を行なう処理である。その過程で、関係テーブル600に登録内容の不足が存在することが検出された場合には、ユーザに新たな三項関係の設定を要求する。処理1407の詳細処理を図15を用いて説明する。

【0088】図15は原テキスト中の仮タグをSGMLタグに変換する処理の詳細フローである。

【0089】処理1501は原テキストから同一文字種が連続した文字列を取り出す処理である。同一文字種が連続した文字列とは、漢字、ひらがな、カタカナ、数字、英字、記号のいずれか一つの種類で構成される文字列である。

【0090】処理1503は処理1501で文字列が取り出せたか否かを判定する処理である。文字列が取り出せない場合には処理1407を終了し、処理1501で文字列が取り出せた場合には処理1505に移る。

【0091】処理1505は処理1501で取り出した文字列が仮タグの開始記号"<<"であるか否かを判定

する処理である。処理1501で取り出した文字列が記号"<<"である場合は処理1507に移り、記号"<<"でない場合は処理1506に移る。

【0092】処理1506は処理1501で取り出した文字列を結果ファイル311に出力する処理である。

【0093】処理1507は仮タグ内のタグ名称を表す文字列を取り出す処理である。また、この処理内で仮タグの種類が開始仮タグであるか終了仮タグであるかの判定を同時に行う。

10 【0094】処理1509は関係テーブル600から処理107または処理119でユーザが指定した文書型定義ファイル名と処理1507で取り出した仮タグ名称を持つ関係を検索する処理である。

【0095】処理1510は処理1509の検索が成功したか否かを判定する処理である。検索が成功したときは処理1511に移り、検索が失敗したときは処理1513に移る。

20 【0096】処理1511は処理1509で検索した関係に含まれるSGMLタグを結果ファイル311に出力する処理である。処理1507で開始仮タグを取り出した場合は開始タグを、処理1507で終了仮タグを取り出した場合は終了タグを出力する。

【0097】処理1513はユーザにSGMLタグ名を要求する処理である。ユーザは目的とする文書型定義のSGMLタグ名を入力する。

30 【0098】処理1515は処理1513で入力されたSGMLタグ名を持つSGMLタグを結果ファイル311に出力し、処理1507で取り出した仮タグと処理107または処理119でユーザが指定した文書型定義ファイル名と処理1513でユーザが入力したSGMLタグ名の組を記憶する処理である。結果ファイルに出力するSGMLタグは処理1507で開始仮タグを取り出した場合は開始タグを、処理1507で終了仮タグを取り出した場合は終了タグとする。

【0099】処理1409はSGMLバーサ305に結果ファイル311中の文書インスタンスを検証させるための処理要求を行う処理である。処理要求は連絡領域303にSGMLバーサ305の処理要求を出力することで行う。

40 【0100】処理1411はSGMLバーサ305の検証結果待ち処理である。SGMLバーサ305の検証結果は連絡領域303にSGMLバーサ305が出力する。

【0101】処理113はSGMLバーサ305で結果ファイル311中の文書インスタンスを検証する処理である。処理113の詳細処理を図16を用いて説明する。

【0102】図16は処理113の詳細フローである。

50 【0103】処理1601は連絡領域303の情報を読み込む処理である。

【0104】処理1603は処理1601で読み込んだ情報が構造化文書作成プログラム301からの検証処理要求か否かを判定する処理である。検証処理要求のときは処理1603に移り、検証処理要求でないときは処理1601に移る。

【0105】処理1605は結果ファイル311から文書インスタンスを読み込み、文書インスタンスの検証を行う処理である。文書インスタンスの検証では、文章型定義に沿ったタグづけが行われているか否かを調べる。

【0106】処理1607は処理1605の検証結果を連絡領域303に出力する処理である。

【0107】処理115は処理113の検証の成否を判定する処理である。検証が成功したときは処理117に移り、検証が失敗したときは構造化文書作成処理を終了する。

【0108】処理117は処理1515で記憶した仮タグとSGMLタグと文書型定義ファイル名称の組を関係テーブル600に格納する。処理117によって新規に格納された関係の例を図7に示す。図7において関係701が処理117によって新たに関係テーブル600に加えられた関係である。関係701は原テキスト中に存在した仮タグのうち図6の関係テーブル600には存在しなかったためユーザが仮タグとSGMLタグとの関係を定義することで関係テーブル600に加えられたものである。

【0109】処理119は文書型定義ファイル名称をユーザに要求する処理である。処理119は処理101で文書型定義の検索に失敗した場合や検索した文書型定義に目的とする文書型定義がない場合にユーザが適当な文書型定義を用いて文書インスタンスの検証を行うための手段となる。

【0110】なお、本実施例では、関係テーブル600の登録内容の不足の検出を処理111におけるタグ変換処理の過程で行なっているが、これを、処理113において変換後の構造化文書の整合性をバーサを用いて検証する際に行なうことも可能である。その場合には、不足を解消する三項関係の入力をユーザに要求し、ユーザが入力した新たな三項関係を考慮したタグ変換処理を行ない、変換結果の整合性をバーサを用いて再検証する。

【0111】

【発明の効果】本発明により、ユーザは自らの扱いやすいユーザ定義タグ情報のみを用いて原テキストを作成すればよく、文書論理構造定義名称や、その定義に従った

記述方法を正確に把握しておく必要はない。そのため、対象とする構造化文書形式を扱い慣れない者にとっても、容易にその文書形式に従った文書インスタンスを作成することが可能になる。

【図面の簡単な説明】

【図1】図1は文書型定義検索プログラムの処理フローについて説明する図である。

【図2】図2はハードウェア構成について説明する図である。

10 【図3】図3はシステム構成について説明する図である。

【図4】図4は原テキストについて説明する図である。

【図5】図5は文書インスタンスについて説明する図である。

【図6】図6は関係テーブルについて説明する図である。

【図7】図7は更新後の関係テーブルについて説明する図である。

20 【図8】図8は仮タグを用いた文書型定義検索の詳細フローについて説明する図である。

【図9】図9は仮タグを用いた文書型定義検索の別法の詳細フローについて説明する図である。

【図10】図10は検索した文書型定義を検索頻度順に表示する処理の詳細フローについて説明する図である。

【図11】図11は文書型定義のファイル名と内容の表示を示す図である。

【図12】図12は文書型定義の選択画面を示す図である。

30 【図13】図13は文書型定義の選択をユーザに要求する処理の詳細フローについて説明する図である。

【図14】図14は原テキストの文書インスタンス変換について説明する図である。

【図15】図15は原テキスト中の仮タグをSGMLタグに変換する処理の詳細フローについて説明する図である。

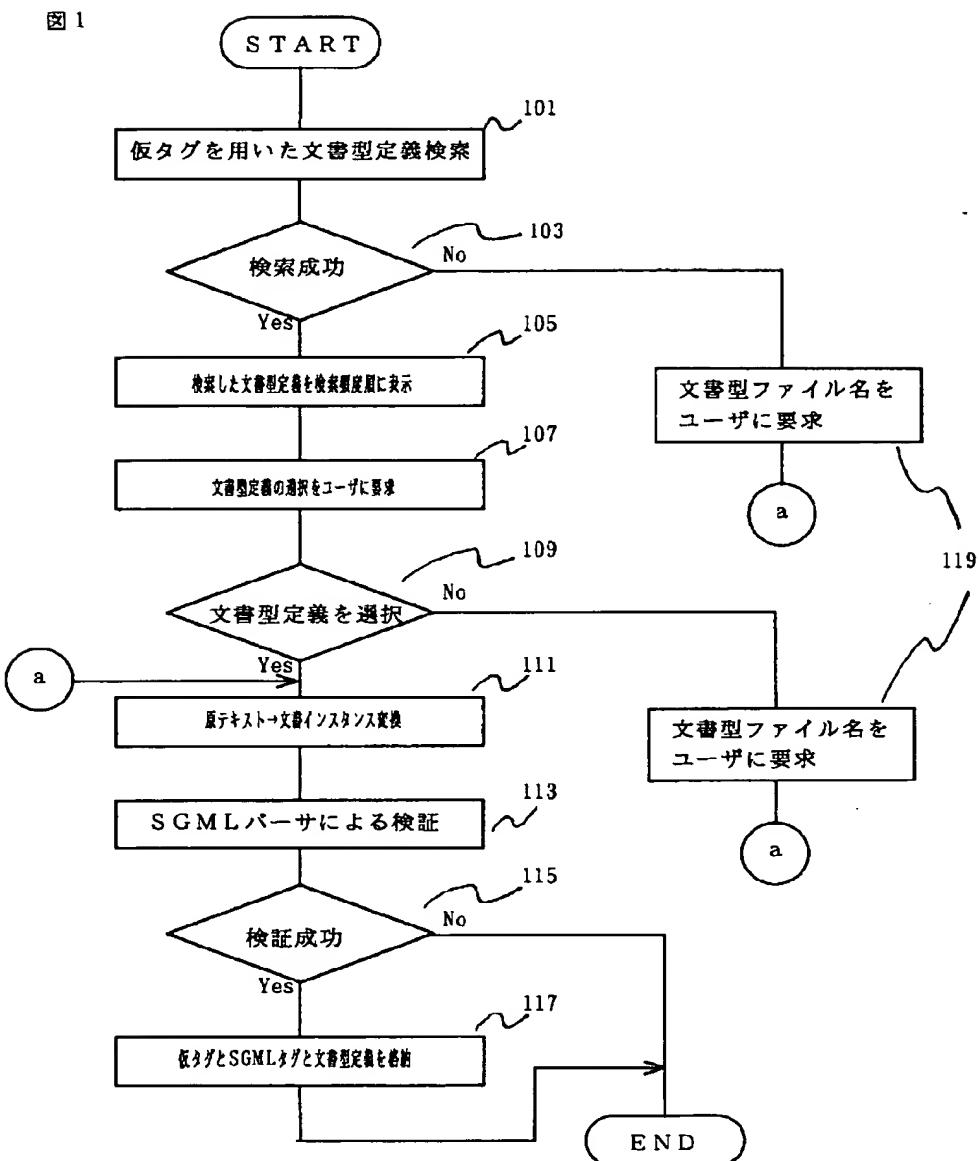
【図16】図16はSGMLバーサによる検証処理について説明する図である。

【符号の説明】

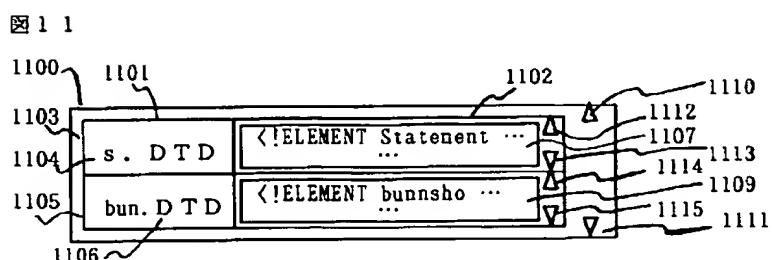
301…構造化文書作成プログラム、305…SGML

40 バーサ、307…原テキストファイル、309…関係テーブルファイル、311…結果ファイル、600…関係テーブル

【図1】

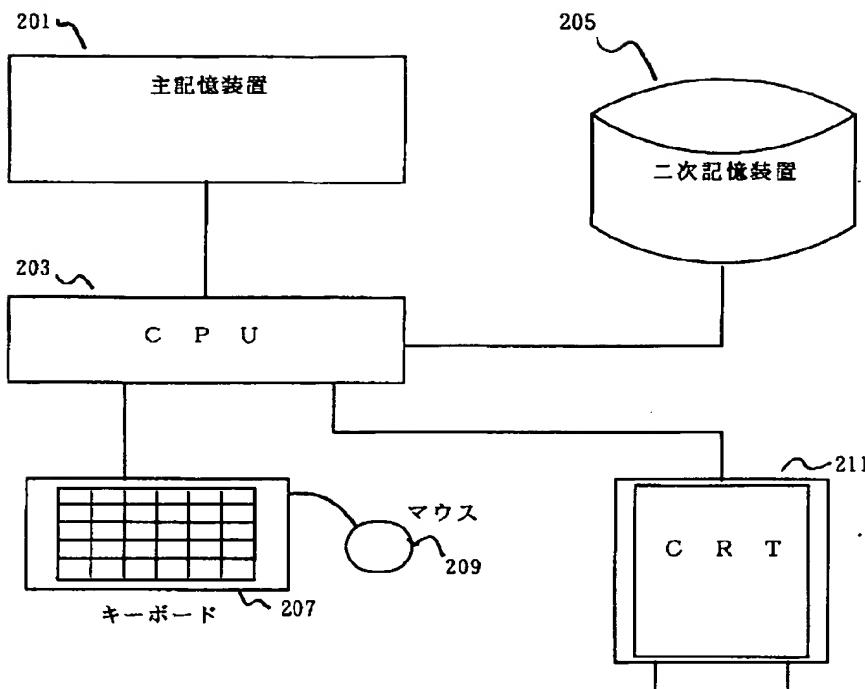


【図11】



【図2】

図2



【図4】

【図5】

図4

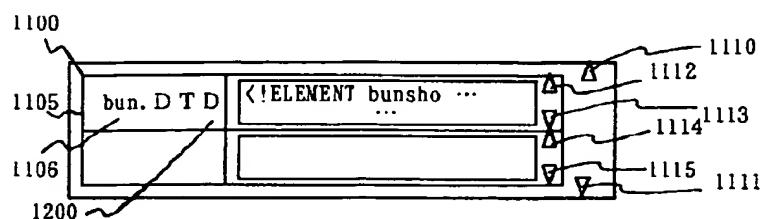
401
 <文>
 <タイトル>発明について</タイトル> 402
 403 ~~~~~ 404
 <1.>問題点</1.> 405
 406 ~~~~~ 407
 <2.>問題点</2.> 408
 409 <参考>特開平××-×××××号公報</参考> 410

図5

501
 !DOCTYPE statement SYSTEM "a.dtd">
 <statement> 502
 503 <title>発明について</title> 504
 505 <chaph>問題点</chaph> 506
 507 <chaph>結果</chaph> 508
 509 <bibh>特開平××-×××××号公報</bibh> 510
 </statement> 511

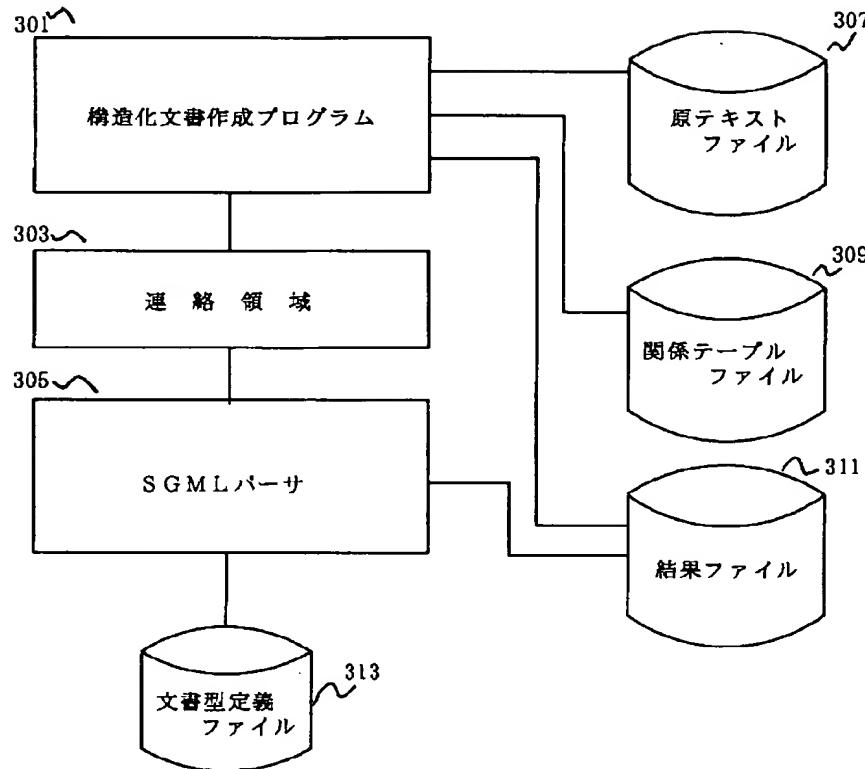
【図12】

図12



【図3】

図3



【図6】

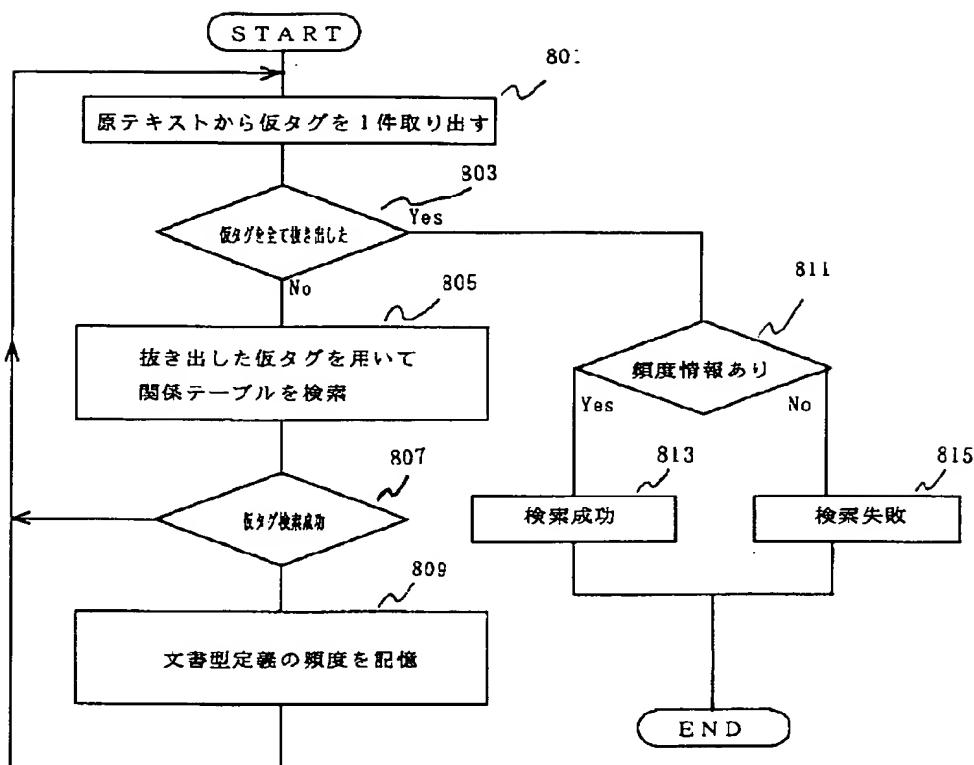
仮タグ	SGMLタグ	文書型定義ファイル名
タイトル	title	s.dtd
1.	chaph	s.dtd
2.	chaph	s.dtd
1.	shou	bun.dtd
2.	shou	bun.dtd

【図7】

仮タグ	SGMLタグ	文書型定義ファイル名
タイトル	title	s.dtd
1.	chaph	s.dtd
2.	chaph	s.dtd
1.	shou	bun.dtd
2.	shou	bun.dtd
参考	bib	s.dtd

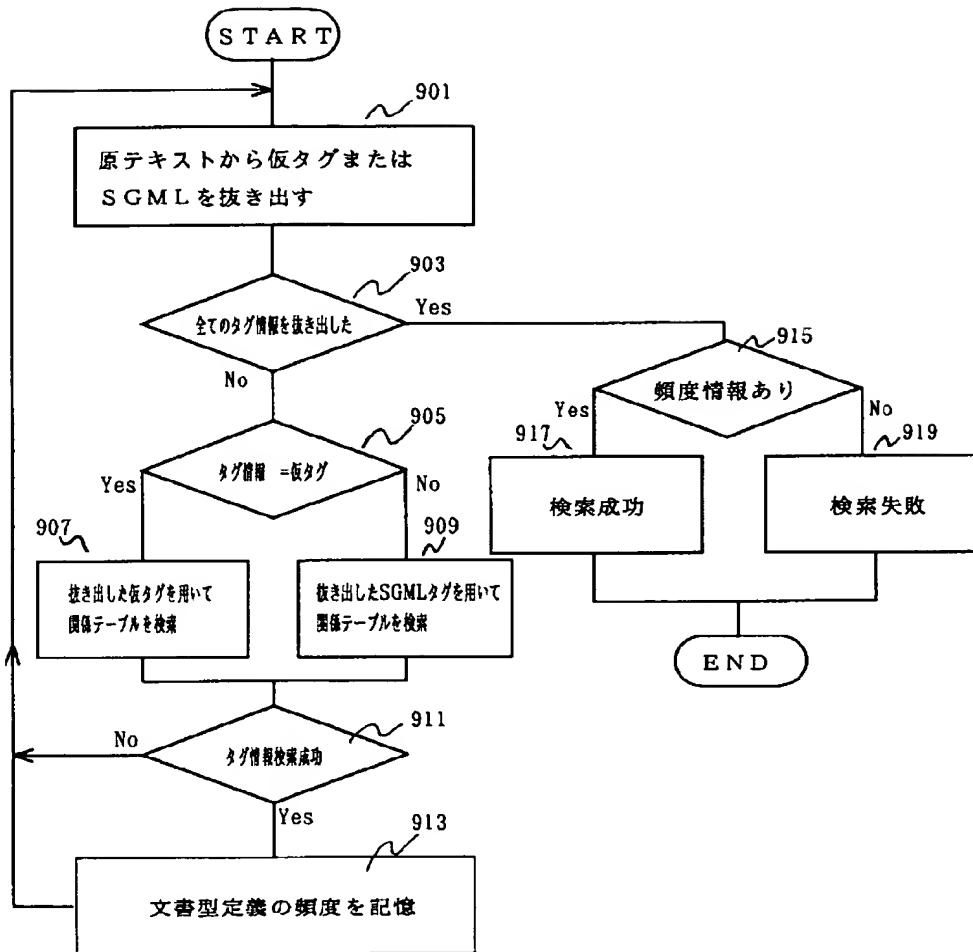
【図8】

図8



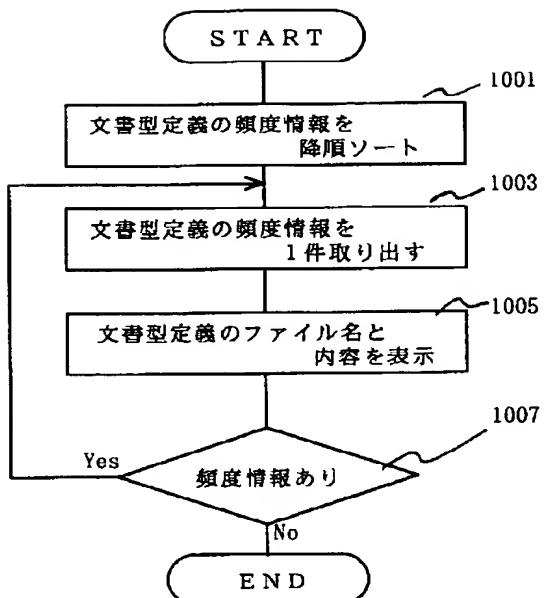
【図9】

図9



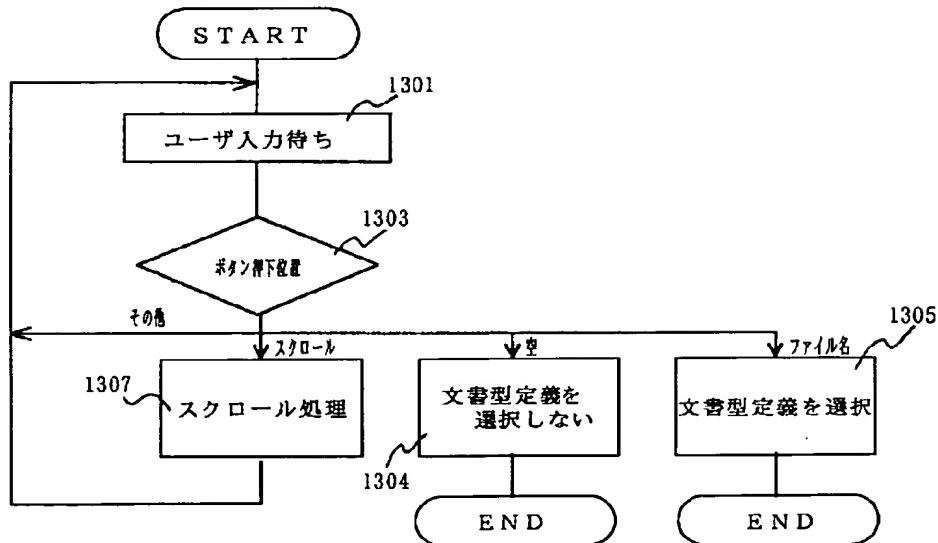
【図10】

図10



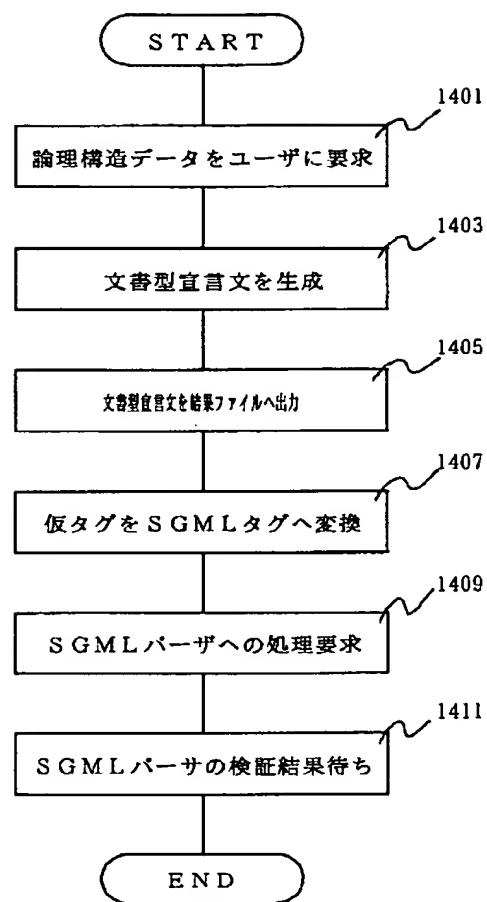
【図13】

図13



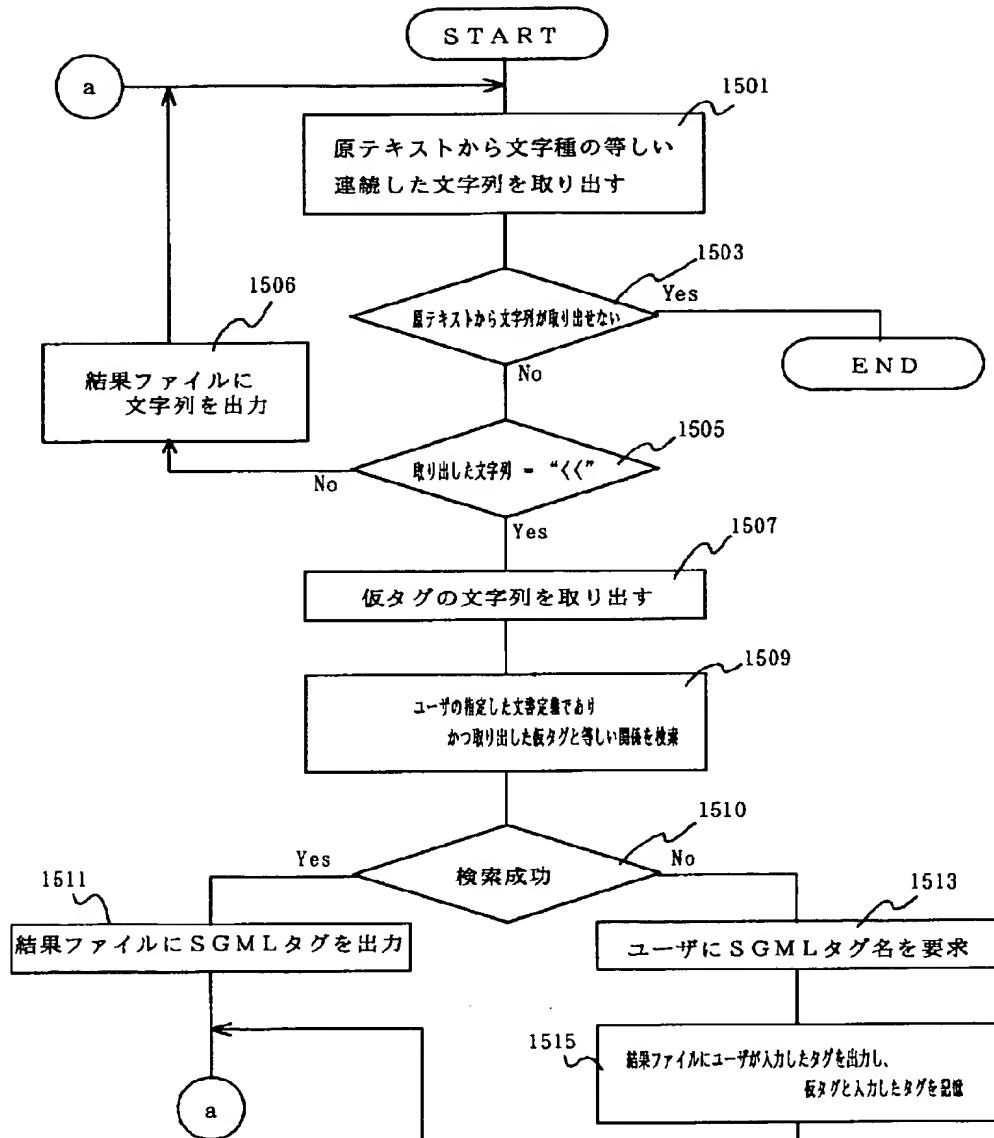
【図14】

図14



【図15】

図15



【図16】

図16

